PROGRAM CODE

```
#include <stdio.h>
#include <stdlib.h>
#define MAX 20
int files[MAX];
struct indarr
{
        int fname;
        struct node *shead;
}p[MAX];
struct node
{
        int *address;
        int index;
        struct node *link;
};
void main()
{
        for(int i=0;i<MAX;i++)
                files[i]=NULL;

        int flen,ibk;
        int fnam,temp,ch;
        new:

        printf("Enter File Name = ");
        scanf("%d",&fnam);

        printf("Enter the file length = ");
        scanf("%d",&flen);

        x:
        printf("Enter the index block = ");
        scanf("%d",&ibk);

        if(files[ibk]!=NULL)
        {
                printf("Index Block Occupied\n");
                goto x;
        }
        else
        {
                files[ibk]=88;
        }

        struct node *head=NULL;

        for(int i=0;i<flen;i++)
        {
                y:
                printf("Enter the (%dth/%d) Block = ",i+1,flen);
```

```
                scanf("%d",&temp);
                if(files[temp]!=NULL)
                {
                        printf("Block(%d) Occupied\n",temp);
                        goto y;
                }
                else
                {
                        files[temp]=fnam;
                        p[ibk].fname=fnam;
                        struct node *new=(struct node *)malloc(sizeof(struct node));
                        new->address=&files[temp];
                        new->index=temp;
                        new->link=NULL;
                        struct node *ptr=head;;
                        if(head==NULL)
                        {
                                head=new;
                                p[ibk].shead=new;
                        }
                        else
                        {
                                while(ptr->link!=NULL)
                                        ptr=ptr->link;
                                ptr->link=new;
                        }
                }
        }
        printf("\nFILE :- %d\tIndex Block :- %d\t",p[ibk].fname,ibk);
        printf("Block (Index -> Value) :- ");
        struct node *ptr2=p[ibk].shead;
        if(head==NULL)
                printf("Empty List\n");
        else
        {
                while(ptr2!=NULL)
                {
                        int tmp=*(ptr2->address);
                        printf("%d->%d , ",ptr2->index,tmp);
                        ptr2=ptr2->link;
                }
        }
        printf("\nDo want to continue ? [no0/yes1] = ");
        scanf("%d",&ch);
        if(ch)
                goto new;
        else
                exit(0);

}
```

OUTPUT


gopikrishna_52@GOPIKRISHNA: ~/Desktop
gopikrishna_52@GOPIKRISHNA:~/Desktop$ gcc linked.c
gopikrishna_52@GOPIKRISHNA:~/Desktop$ ./a.out
Enter File Name = 3
Enter the file length = 3
Enter the index block = 3
Enter the (1th/3) Block = 3
Block(3) Occupied
Enter the (1th/3) Block = 4
Enter the (2th/3) Block = 5
Enter the (3th/3) Block = 5
Block(5) Occupied
Enter the (3th/3) Block = 6
FILE :- 3 Index Block :- 3 Block (Index -> Value) :- 4->3 , 5->3 , 6->3 ,
Do want to continue ? [no0/yes1] = 1
Enter File Name = 2
Enter the file length = 2
Enter the index block = 2
Enter the (1th/2) Block = 5
Block(5) Occupied
Enter the (1th/2) Block = 7
Enter the (2th/2) Block = 12
FILE :- 2 Index Block :- 2 Block (Index -> Value) :- 7->2 , 12->2 ,
Do want to continue ? [no0/yes1] = 0
gopikrishna_52@GOPIKRISHNA:~/Desktop$